TU Clausthal

Clausthal University of Technology

Where Logic and Agents Meet[1]

Jürgen Dix and Michael Fisher

IfI Technical Report Series IfI-10-08

IfI

Department of Informatics
Clausthal University of Technology

Impressum

Publisher: Institut für Informatik, Technische Universität Clausthal
Julius-Albert Str. 4, 38678 Clausthal-Zellerfeld, Germany
Editor of the series: Jürgen Dix
Technical editor: Michael Köster
Contact: michael.koester@tu-clausthal.de
URL: http://www.in.tu-clausthal.de/forschung/technical-reports/
ISSN: 1860-8477

The IfI Review Board

Where Logic and Agents Meet[†]

Jürgen Dix and **Michael Fisher**

Clausthal University of Technology, Germany and University of Liverpool, United Kingdom

Abstract

Multi agent systems have been invented 20 years ago and the theory has matured in the last two decades. Logic has and still is playing a prominent part in the basic foundations of agency and also in the development of agent programming languages, the specification and verification of agent systems. This paper is a (subjective) overview of the points of contact of logic and agents as the authors perceived it over the years.

1 Introduction

The notion of an 'agent' came about through the confluence of several important ideas: *distributed computation*; *object-based systems*; Artificial Intelligence techniques, such as *planning* and *learning*; and ideas from Philosophy about *rational action*, *choice* and *commitment*. The basic idea is of a *truly* autonomous computational entity, able to make its own choices about targets, tasks, and plans. Yet, as the area developed, arguments about what exactly should constitute an agent [139, 72] led on to more sophisticated definitions of a *rational agent* [140], essentially excluding autonomous computational systems based on neural networks or other, primarily stochastic, foundations. Our view of a 'rational agent' is as a distributed, autonomous, computational entity with its own goals, information and ways of making choices. When such an agent executes it will do so with some goal in mind but can, at any moment in time, decide to change between goals or between ways of achieving that goal [137]; such changes come about as the result of *deliberation* [37, 47].

*This work was partly funded by the NTH School for IT Ecosystems. NTH (Niedersächsische Technische Hochschule) is a joint university consisting of Technische Universität Braunschweig, Technische Universität Clausthal, and Leibniz Universität Hannover.

†This work was partly funded by the NTH School for IT Ecosystems. NTH (Niedersächsische Technische Hochschule) is a joint university consisting of Technische Universität Braunschweig, Technische Universität Clausthal, and Leibniz Universität Hannover.

The use of one agent on its own can be interesting, but it is only when we consider *multiple* agents, or *multi-agent systems* that the full power of the agent metaphor becomes clear. As we move to a multi-agent scenario, not only must the agents themselves now be able to communicate, but it makes sense for each agent to have views on the other agents. Thus, while an agent typically has goals and information about itself, in a multi-agent context it can have goals about the activities of the multi-agent systems and knowledge about other agents in the system. This leads on to very sophisticated *interaction*, *coordination*, and *cooperation* within a multi-agent system. If we are to describe or analyze such multi-agent systems then we must surely take account of the *social* and well as the *individual* behaviours of the agents.

The multi-agent paradigm has become increasingly popular and is now the metaphor of choice within complex systems. Its popularity is to do both with the simplicity of the ideas, and with its ability to capture a very wide range of different aspects — not just functional, but societal, emotional, and economic. This, together with the vast array of languages, frameworks and tools now available means that practical construction of multi-agent systems can go together with theoretical analysis.

In this article we provide a short, and very subjective, description of how mathematical techniques, primarily those from logical foundations, have been used in multi-agent systems. This is certainly *not* a comprehensive survey, but gives some indication of where the fields of "Logic" and "Agents" have met in recent years. The Annals of Math and AI contain a surprising number of articles in this area, among them several special issues focussing on computational logic and its use in agent systems.

2 Describing Agents

How shall we describe an 'agent'? As we commented above, to some extent we can see an agent as a distributed component, an object, an AI subsystem, or as a philosophical artifact. Indeed, some people have argued that there is nothing special about the agent view. By contrast, we contend that, by considering all these aspects together, the agent metaphor provides a high-level and holistic view of quite complex processes that cannot easily be seen just as distributed computation, object execution, etc.

A key development in describing and understanding agents, particularly rational agents, was the introduction of the BDI (Belief, Desires, Intentions) view [114]. This was based on both Bratman's philosophical work on intentions [27] and on experiences with practical AI planning [73]. The BDI view contains two aspects: a theoretical framework, comprising distinguished *beliefs*, *desires*, and *intentions* together with the interactions between them [114, 113]; and a practical architecture providing a basis for agent-based implementations [115, 116]. Here we will mainly address the first of these, but note that most agent systems, including the languages and frameworks mentioned later, are based on the ideas from the BDI architecture.

There have been, of course, logical techniques used for agents that are different to the BDI approach, with some of the most influential being [91, 43, 44, 100]. However, we will particularly focus on the BDI approach.

Recall that, in the introduction, we said that the core elements of a rational agent were some goals the agent has, some information it has, and some deliberation techniques for switching between goals based on this information. The BDI approach refines this a little further to incorporate:

- *beliefs* — describing the agent's view of itself, other agents and its environment (note that the term 'belief' is used rather than 'knowledge' since the agent can not be *certain* of its information and it can only *believe* this information about the world and other agents);
- *desires* — describing the agent's long term goals; and
- *intentions* — describing those long term goals that have been selected and are actively being pursued.

Thus,

> *I believe I have one hour of uninterrupted time available, my desire is that eventually this article will be finished and published, and my immediate intention is to write this particular section.*

In the BDI architecture the above sets are usually supplemented by a set of *actions* describing the primitive ways the agent can affect the world, and a set of *plans* describing how the agent might achieve certain goals. We will see later that these play a significant part in practical agent programming but, for the moment, we will concentrate on beliefs, desires, and intentions.

In the logical formalisation of BDI agents, a key development was the use of various modal and temporal logics for this purpose [113, 121]. Thus, in describing agents we (at least) need some logical formalism for capturing the underlying *dynamic* nature of an agent, and then some (possibly different) formalism for capturing the BDI aspects. Typically, dynamic [80, 130] or temporal [65] logics have been used for the first of these. For the BDI aspects, the following are used.

1. A logic of belief, typically **KD45** modal logic [16], but alternatively fuzzy logics, logics of probabilistic belief, or multi-context logics [76, 63], or even (if we are certain) logics of knowledge [66].
2. A logic of desires, typically **KD** modal logic, and
3. A logic of intentions, again a **KD** modal logic.

Importantly there are many interactions between these dimensions, for example[1]

$$\neg(\mathsf{I}\varphi \wedge \neg\mathsf{D}\varphi)$$

meaning that the agent should never intend to do something that it does not explicitly desire. These logical dimensions, together with the underlying temporal or dynamic basis, allow us to describe quite complex agent activities. For example

$$\mathsf{I}\mathit{write\_section} \wedge \Box \mathit{can\_write} \quad \Rightarrow \quad \mathsf{B}\Diamond \mathit{finished\_section}$$

might be interpreted as

> *"if I intend to write the section and can always write, then I believe that eventually the section will be finished".*

We can see from this that agent descriptions can become quite complex and indeed, since the logical basis is a combination [94] of temporal and (several) modal logics, the formal analysis of such descriptions (see later) can be quite expensive.

Before we conclude this section, we just note that, while the BDI approach is widely used, there are other options. One interesting one is the KARO [131] framework (where KARO stands for "Knowledge, Abilities, Results, and Opportunities"). Again, this is based on the idea of having an underlying *dynamic* component together with both *informational* and *motivational* components, but with a different approach to that of BDI. Thus, the KARO approach combines dynamic aspects via propositional dynamic logic, informational aspects via **S5** modal logic, and motivational aspects via **KD** modal logic.

There are numerous other approaches, including those involving linear logic [81], dynamic logic [122], the classical situation calculus [78], and the event calculus [142].

A final aspect is that, when we move towards describing an agent's environment we might well have to involve real-time [4], spatial [45] or probabilistic [79] logics.

3 Agent Interactions and Group Dynamics

As opposed to a single agent system, we face new and difficult challenges in a *multi-agent* system. In this section we briefly consider two of the most important challenges and list some significant developments in logical approaches to tackling them. The challenges are:

[1]From now on: I means "I intend"; D means "I desire"; B means "I believe"; $\Box$ means "always"; and $\Diamond$ means "sometimes".

- How do agents interact with each other?
- What goals can groups of agents achieve?

Agents have to talk to each other and therefore they need a (standardized) language to do so. Some popular approaches for agent communication are **KQML**, **KIF**, and **Fipa-ACL**, which have their roots in the *speech act* theory of Searle from linguistics [124]. These approaches come from knowledge representation and the part of Philosophy that is closely related to formal logic [127, 134, 101, 77].

Agents developed within one multi-agent system should be able to deal with agents from other systems as well. As one cannot assume that they all have the same "background" (knowledge about the world), they need to have access to a knowledge base where the fundamental concepts they talk about are defined. This is where the idea of the semantic web and *ontologies* comes in. Again, this approach is perfectly suited for logic-based methodologies and many papers have been devoted to deal with it [110, 84, 8, 11] in addition to the vast amount of publications on ontologies and description logics in general [9].

We mention in passing another important part of interaction, namely *negotiation*: how to reach a satisfactory conclusion among self-interested agents? It turns out that many techniques from logic, *argumentation* and *game theory* can be successfully applied here: [60, 6, 93, 35, 29, 28]

While the notion of an agent as an autonomous entity is a very important one, the full power of the agent paradigm is only obtained by the interplay of multiple agents. Often, a single agent is not able to bring about a more complicated goal. Several agents may be able to reach this goal, but their success often depends on how the remaining agents behave. This leads to various branches in multi-agent systems, e.g. *coalition formation* (which coalitions emerge and how do they change over time?) and *what can coalitions bring about* (how can one describe and determine the behaviour of teams of agents?). Coalition formation typically depends on notions of common goals, shared intentions, and the agents' level of knowledge about their environment. While the development of an agent system over time can be described by temporal logics, such as **LTL** [108], and **CTL** [64, 65, 41, 39], these logics are not appropriate to succinctly express the strategic behaviour of agents. For example:

> *is there a strategy for some team of agents to make sure that a certain property holds?*

To encompass such requirements, standard temporal logics have been extended with game-theoretic concepts [123]. Thus, the logics **ATL*** and **ATL** [5] (*Alternating-time Temporal Logic*) and various extensions of them are logics of this kind. They contain *cooperation modalities* of the form $\langle\langle A \rangle\rangle$ where 'A' is a team of agents. The formula $\langle\langle A\rangle\rangle\gamma$ expresses the statement that team A has a *collective strategy* that can enforce γ. The complexity of these logics, in particular the model checking problem, has been studied in detail [30, 33, 87, 129, 95]. Various extensions of **ATL** which allow us to express well-known solution concepts from game theory, such as Nash equilibria, sub-game perfect Nash equilibria, etc., have also been developed [34] and, recently, versions of **ATL** describing resource-bounded agents have also been defined [32, 31].

Finally, there is the vast area of work covering *norms*, *institutions* and *organisations* in agent systems. The idea here is to look at how human societies have coped with decentralized control, coordination and cooperation, and to re-use such structures in multi-agent systems. Norms and conventions within human societies have evolved over time and research in this area tries to model the appropriate abstractions for multi-agent systems. In this area, we mainly highlight the workshop series devoted to this branch of agent systems [106, 107, 126].

4 Agent Programming

While the modelling and analysis of systems in terms of agents has many benefits, particularly for clarity and conciseness, it is important to have appropriate programming abstractions and robust languages in which agent-based systems can be implemented. While in some cases agents are implemented via the use of traditional programming languages it is increasingly clear that the novel aspects of agents require suitably novel programming languages.

Just as the concept of an 'agent' evolved from several different areas, so agent programming frameworks show a similarly diverse background [18, 55, 19]. However, the main development began with AI languages such as PROLOG. Recall that, in PROLOG, we have an ordered list of 'goals', together with an ordered list of rules and facts. These rules/facts are searched in a sequential manner in order to try to reduce the first goal on the list to nothing (via a 'fact') or to further sub-goals (via a 'rule'). If goal reduction fails then the execution backtracks to a previous choice point and continues. Many rational agent languages work in a similar way. However, a key aspect of an agent is that it can react to new information (e.g. sensor data) and modify its current goals. So, the PROLOG execution mechanism is essentially modified in two ways:

1. the current list of 'goals' can be re-ordered or modified at any moment — since computation concerns reducing the first goal on the list, this change of emphasis can be viewed as the deliberation the agent undertakes about its current goal [67]; and

2. the list of facts and rules can be re-ordered — since this ordering describes the next rule/fact to be attempted, then *re-ordering* corresponds to the agent deliberating about, and changing its view of, *how* to reduce the current goal.

There is, of course, much more to these languages but the above provides a general framework. In BDI based languages the 'facts' the agent has are expanded to include quite complex beliefs and the goals the agent works towards are refined to be 'intentions'. The key language in this areas is AGENTSPEAK [112] developed to be a version of standard declarative programming adapted to the BDI framework. This has led on to very many interesting and useful languages of a similar form, such as JASON [24, 25], 3APL, [52], and GOAL [83]. Particularly the first two of these have been used in many practical applications.

As mentioned above, agent languages also evolved from many areas. So there are:

- extensions of `Java` with BDI architectures [136, 109, 102];
- agent extensions of *logic programming* [97, 92, 38, 3, 119, 133, 1, 17] and *answer set programming* [86, 42, 132, 46];
- languages providing an interaction layer for legacy systems [128, 62, 54]; and
- languages based on executable specification of concurrent agents such as METATEM [69, 68] and [10].

There are increasingly many such languages though important work is now taking place assessing and evaluation the different approaches from both theoretical [15] and practical [12] viewpoints.

5 Agent Verification

As the agent, and particularly *multi-agent*, metaphor is used in more sophisticated and critical scenarios, so attention has turned to questions concerning the *reliability* and *predictability* of such systems. Multi-agent systems are used in many such scenarios, for example automated stock trading systems [82, 125], process control [120], space exploration [103, 36], sensor networks [96, 118], health-care systems [141], air traffic systems [105, 104], etc [88]. While multi-agent systems are widely deployed there is often a lack of trust among users that the agents will *always* work as expected in critical scenarios.

Unsurprisingly, therefore, there has been a move towards fully formal justification of the reliability of these agents [70]. As within standard Computer Science, there is a wide array of techniques that can potentially be adapted for agent verification [117]. Here, some formal (usually logical) description of the required behaviour of a system, say Req, is checked against some representation of the system. In *deductive verification*, a logical description of the system is given, say Sys, and then proof methods are used to establish that $Sys \Rightarrow Req$. An alternative approach is to use *algorithmic verification* whereby Req is checked on all possible routes through some structure describing the behaviour of the system. In *model checking* [40], the structure explored is usually a finite-state automaton representing all the runs/executions of the system. This approach is certainly the most popular and successful variety of formal verification.

Mirroring the work of formal verification in other areas, the verification of multi-agent systems is indeed a very active research area [21, 70, 51]. Again, although deductive and rewriting approaches have been developed [71, 85, 2, 53, 1, 7, 26] it is the techniques based on *model-checking* that have been particularly popular [14, 21, 138, 75, 13, 74, 90, 87, 89]. Work in this area has led to a variety of different practical agent verification systems. One class verifies the required property against a model (structure, as described above) representing possible agent behaviours; a leading example of this is MCMAS [98, 111, 99]. An alternative approach, based on the idea of *program model checking* [135], involves directly verifying the agent program code rather than a model of its executions. A leading example of this style of agent verification is MCAPL/AJPF [22, 20, 23].

6 Where to now?

Logic (particularly computational logic) continues to be central to the area of multi-agent systems: logical approaches are vital for the specification and semantics of such systems; programming metaphors based on computational logic are predominant amongst agent languages; and formal verification techniques are increasingly used to assess the safety and reliability of multi-agent systems. All of these areas look set to increase in activity and popularity.

The research community associated with these areas is particularly active. For example, the 11th international workshop on *Computational Logic in Multi-Agent Systems* (CLIMA) was held in 2010; the 12th edition will happen in 2011. CLIMA (`http://centria.di.fct.unl.pt/~clima`) brings together researchers tackling all the areas covered within this article, together with many more [56, 57]. The *Annals of Mathematics in Artificial Intelligence* has published special issues related to some of these CLIMA workshops, for example [61, 58], as well as special issues related to other activities linking logic and agents, for example [55, 70].

On the practical side, many tools, techniques and languages have been developed and an interesting and useful initiative involves the *multi-agent programming contest*. This started in [48], and continues today [49, 50, 12]. The contest aims to provide an objective, and non-trivial, problem to which various different agent programming languages can be applied. This has led to a deeper understanding of the most appropriate ways to program multi-agent systems as well as, of course, improvements in the languages themselves.

As we have seen, logic and agents have a natural and productive interaction. Activity in this area is not only set to continue, but increase. Formal specification, automated synthesis, and deductive verification, via logics of agency are sure to come into their own. Agent programming languages are about to become *mainstream*. Formal agent verification techniques are essential for trusted, mobile agents. These involve not only the model-checking techniques mentioned above, but other lightweight techniques such as runtime verification. Negotiation, cooperation and planning [59] are all required within the increasingly sophisticated agent applications. And so on.

In summary, although we can only provide a brief (subjective) view of the field in this article, we hope the reader has finished it with a strong and positive feel for the area "where logic and agents meet".

References

[1] M. Alberti, F. Chesani, M. Gavanelli, E. Lamma, P. Mello, and P. Torroni. Verifiable Agent Interaction in Abductive Logic Programming: The SCIFF framework. *ACM Transactions on Computational Logic*, 9(4), 2008.

[2] N. Alechina, M. Dastani, B. Logan, and J. Meyer. A Logic of Agent Programs. In *Proc. 22nd American National Conference on Artificial Intelligence (AAAI)*, pages 795–800. AAAI press, 2007.

[3] J.-J. Alferes, A. Brogi, J. A. Leite, and L. M. Pereira. Logic Programming for Evolving Agents. In *Proc. 7th International Workshop on Cooperative Information Agents (CIA)*, volume 2782 of *LNCS*, pages 281–297. Springer, 2003.

[4] R. Alur and T. A. Henzinger. Logics and Models of Real Time: A Survey. In *Proc. REX Workshop on Real-Time: Theory in Practice*, volume 600 of *LNCS*, pages 74–106. Springer, 1991.

[5] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-Time Temporal Logic. *Journal of the ACM*, 49:672–713, 2002.

[6] L. Amgoud and C. Cayrol. A Reasoning Model Based on the Production of Acceptable Arguments. *Annals of Mathematics and Artificial Intelligence*, 34(1-3):197–215, 2002.

[7] L. Astefanoaei, M. Dastani, J.-J. Meyer, and F. S. de Boer. On the Semantics and Verification of Normative Multi-Agent Systems. *Journal of Universal Computer Science*, 15(13):2629–2652, 2009.

[8] A. P. Ayala. Ontology Agents and Their Applications in the Web-Based Education Systems: Towards an Adaptive and Intelligent Service. In *Intelligent Agents in the Evolution of Web and Applications*, volume 167 of *Studies in Computational Intelligence*, pages 249–278. Springer, 2009.

[9] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[10] M. Baldoni, A. Martelli, V. Patti, and L. Giordano. Programming Rational Agents in a Modal Action Logic. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):207–257, 2004.

[11] N. Bassiliades, G. Governatori, A. Paschke, and J. Dix. Guest Editors' Introduction: Rule Representation, Interchange, and Reasoning in Distributed, Heterogeneous Environments. *IEEE Transaactions on Knowledge and Data Engineering*, 22(11):1489–1491, 2010.

[12] T. M. Behrens, M. Dastani, J. Dix, and P. Novák. Agent Contest Competition: 4th Edition. In *Proc. 6th International Workshop on Programming Multi-Agent Systems, (ProMAS)*, volume 5442 of *LNCS*, pages 211–222. Springer, 2009.

[13] T. M. Behrens and J. Dix. Model Checking Multi-Agent Systems with Logic Based Petri Nets. *Annals of Mathematics and Artificial Intelligence*, 51(2-4):81–121, 2007.

[14] M. Benerecetti, F. Giunchiglia, and L. Serafini. Model Checking Multiagent Systems. *Journal of Logic and Computation*, 8(3):401–423, 1998.

[15] M. Birna van Riemsdijk, M. Dastani, and J.-J. Meyer. Goals in Conflict: Semantic Foundations of Goals in Agent Programming. *Autonomous Agents and Multi-Agent Systems*, 18(3):471–500, 2009.

[16] P. Blackburn, J. van Benthem, and F. Wolter, editors. *Handbook of Modal Logic*. Elsevier, 2006.

[17] P. A. Bonatti, S. Kraus, and V. S. Subrahmanian. Secure Agents. *Annals of Mathematics and Artificial Intelligence*, 37(1-2):169–235, 2003.

[18] R. H. Bordini, M. Dastani, J. Dix, and A. El Fallah-Seghrouchni, editors. *Multi-Agent Programming: Languages, Platforms and Applications*. Springer, 2005.

[19] R. H. Bordini, M. Dastani, J. Dix, and A. El Fallah-Seghrouchni, editors. *Multi-Agent Programming: Languages, Tools and Applications*. Springer, 2009.

[20] R. H. Bordini, L. A. Dennis, B. Farwer, and M. Fisher. Automated Verification of Multi-Agent Programs. In *Proc. 23rd IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 69–78. IEEE Computer Society Press, 2008.

[21] R. H. Bordini, M. Fisher, W. Visser, and M. Wooldridge. Model Checking Rational Agents. *IEEE Intelligent Systems*, 19(5):46–52, 2004.

[22] R. H. Bordini, M. Fisher, W. Visser, and M. Wooldridge. Verifying Multi-Agent Programs by Model Checking. *Journal of Autonomous Agents and Multi-Agent Systems*, 12(2):239–256, 2006.

[23] R. H. Bordini, M. Fisher, M. Wooldridge, and W. Visser. Property-based Slicing for Agent Verification. *Journal of Logic and Computation*, 19(6):1385–1425, 2009.

[24] R. H. Bordini, J. F. Hübner, and R. Vieira. *Jason* and the Golden Fleece of Agent-Oriented Programming. In Bordini et al. [18], chapter 1, pages 3–37.

[25] R. H. Bordini, J. F. Hübner, and M. Wooldridge. *Programming Multi-agent Systems in AgentSpeak Using Jason*. Wiley, 2007.

[26] R. H. Bordini and Á. F. Moreira. Proving BDI Properties of Agent-Oriented Programming Languages. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):197–226, 2004.

[27] M. E. Bratman. *Intentions, Plans, and Practical Reason*. Harvard University Press, 1987.

[28] N. Bulling and J. Dix. Modelling and Verifying Coalitions using Argumentation and ATL. *Inteligencia Artificial*, 14(46):45–73, 2010.

[29] N. Bulling, J. Dix, and C. Chesñevar. An Argumentative Approach for Modelling Coalitions using ATL. In *Proc. of the 5rd. International Workshop on Argumentation and Multiagent Systems (ArgMAS)*, volume 5384 of *LNCS*, pages 190–211. Springer, 2009.

[30] N. Bulling, J. Dix, and W. Jamroga. Model Checking Logics of Strategic Ability: Complexity. In Dastani et al. [51].

[31] N. Bulling and B. Farwer. Expressing Properties of Resource-Bounded Systems: The Logics RBTL and RBTL*. In Dix et al. [56].

[32] N. Bulling and B. Farwer. On the (Un-)Decidability of Model-Checking Resource-Bounded Agents. In *Proc, 19th European Conference on Artificial Intelligence (ECAI)*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 567–572. IOS Press, 2010.

[33] N. Bulling and W. Jamroga. What Agents Can Probably Enforce. *Fundamenta Informaticae*, 93:81–96, 2009.

[34] N. Bulling, W. Jamroga, and J. Dix. Reasoning about Temporal Properties of Rational Play. *Annals of Mathematics and Artificial Intelligence*, 53(1-4):51–114, 2009.

[35] A. Ciampolini, E. Lamma, P. Mello, F. Toni, and P. Torroni. Cooperation and Competition in ALIAS: A Logic Framework for Agents that Negotiate. *Annals of Mathematics and Artificial Intelligence*, 37(1-2):65–91, 2003.

[36] W. Clancey, M. Sierhuis, C. Kaskiris, and R. van Hoof. Advantages of Brahms for Specifying and Implementing a Multiagent Human-Robotic Exploration System. In *Proc. 16th International Florida Artificial Intelligence Research Society Conference (FLAIRS)*, pages 7–11. AAAI Press, 2003.

[37] W. J. Clancey. Simulating Activities: Relating Motives, Deliberation, and Attentive Coordination. *Cognitive Systems Research*, 3(3):471–499, 2002.

[38] K. L. Clark and F. G. McCabe. Go! - A Multi-Paradigm Programming Language for Implementing Multi-Threaded Agents. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):171–206, 2004.

[39] E. Clarke, E. Emerson, and A. Sistla. Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications. *ACM Transactions on Programming Languages and Systems*, 8(2):244–263, 1986.

[40] E. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.

[41] E. M. Clarke and E. A. Emerson. Design and Synthesis of Synchronization Skeletons Using Branching Time Temporal Logic. In *Proc. Workshop on Logics of Programs*, volume 131 of *LNCS*, pages 52–71, 1981.

[42] O. Cliffe. Specifying and Analyzing Institutions in Multi-Agent Systems using Answer Set Programming. *Knowledge Engineering Review*, 24(4):411, 2009.

[43] P. R. Cohen and H. J. Levesque. Intention is Choice with Commitment. *Artificial Intelligence*, 42:213–261, 1990.

[44] P. R. Cohen and H. J. Levesque. Rational Interaction as the basis for Communication. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 221–256. The MIT Press, 1990.

[45] A. G. Cohn and J. Renz. Qualitative Spatial Representation and Reasoning. In F. van Harmelen, B. Porter, and V. Lifschitz, editors, *Handbook of Knowledge Representation*, chapter 13, pages 551–596. Elsevier Press, 2007.

[46] S. Costantini. Integrating Answer Set Modules into Agent Programs. In *Proc. 10th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, volume 5753 of *LNCS*, pages 613–615. Springer, 2009.

[47] M. Dastani, F. Dignum, and J.-J. C. Meyer. Autonomy and Agent Deliberation. In *Agents and Computational Autonomy — Potential, Risks, and Solutions*, volume 2969 of *LNCS*, pages 114–127. Springer, 2004.

[48] M. Dastani, J. Dix, and P. Novák. The First Contest on Multi-agent Systems Based on Computational Logic. In *Proc. 6th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 3900 of *LNCS*, pages 373–384. Springer, 2006.

[49] M. Dastani, J. Dix, and P. Novák. The Second Contest on Multi-Agent Systems Based on Computational Logic. In *Proc. 7th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 4371 of *LNCS*, pages 266–283, 2006.

[50] M. Dastani, J. Dix, and P. Novák. Agent Contest Competition: 3rd Edition. In *Proc. 5th International Workshop on Programming Multi-Agent Systems (ProMAS)*, volume 4908 of *LNCS*, pages 221–240. Springer, 2008.

[51] M. Dastani, K. V. Hindriks, and J.-J. Meyer, editors. *Specification and Verification of Multi-Agent Systems*. Springer, 2010.

[52] M. Dastani, M. B. van Riemsdijk, and J.-J. C. Meyer. Programming Multi-Agent Systems in 3APL. In Bordini et al. [18], chapter 2, pages 39–67.

[53] F. S. de Boer, K. V. Hindriks, W. van der Hoek, and J.-J. C. Meyer. A Verification Framework for Agent Programming with Declarative Goals. *J. Applied Logic*, 5(2):277–302, 2007.

[54] J. Dix. A Computational Logic Approach to Heterogenous Agent Systems. In *Proc. 6th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, volume 2173 of *LNCS*, pages 1–20. Springer, 2001.

[55] J. Dix, M. Fisher, H. J. Levesque, and L. Sterling. Introduction: Special Issue on Logic-Based Agent Implementation. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):131–133, 2004.

[56] J. Dix, M. Fisher, and P. Novak, editors. *Proc. 10th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA X)*, volume 6214 of *LNAI*, Berlin, 2010. Springer.

[57] J. Dix, J. Leite, G. Governatori, and W. Jamroga, editors. *Proc. 11th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 6245 of *LNCS*. Springer, 2010.

[58] J. Dix, J. A. Leite, and K. Satoh. Introduction. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):1–3, 2004.

[59] J. Dix, H. Muñoz-Avila, D. S. Nau, and L. Zhang. IMPACTing SHOP: Putting an AI Planner Into a Multi-Agent Environment. *Annals of Mathematics and Artificial Intelligence*, 37(4):381–407, 2003.

[60] J. Dix, S. Parsons, H. Prakken, and G. R. Simari. Research Challenges for Argumentation. *Computer Science - R&D*, 23(1):27–34, 2009.

[61] J. Dix, F. Sadri, and K. Satoh. Introduction. *Annals of Mathematics and Artificial Intelligence*, 37(1-2):1–3, 2003.

[62] J. Dix and Y. Zhang. IMPACT: A Multi-Agent Framework with Declarative Semantics. In Bordini et al. [18], chapter 3, pages 69–94.

[63] A. F. Dragoni. Mental States as Multi-Context Systems. *Annals of Mathematics and Artificial Intelligence*, 54(4):265–292, 2008.

[64] E. Emerson and J. Halpern. "Sometimes" and "Not Never" Revisited: On Branching versus Linear Time Temporal Logic. *Journal of the ACM*, 33(1):151–178, 1986.

[65] E. A. Emerson. Temporal and Modal Logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, pages 996–1072. Elsevier, 1990.

[66] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge*. MIT Press, 1996.

[67] M. Fisher. Agent Deliberation in an Executable Temporal Framework. (To appear in *Journal of Applied Logic*.), 2011.

[68] M. Fisher and C. Ghidini. Executable Specifications of Resource-Bounded Agents. *Journal of Autonomous Agents and Multi-Agent Systems*, 21(3):368–396, 2010.

[69] M. Fisher and A. Hepple. Executing Logical Agent Specifications. In Bordini et al. [19], pages 1–27.

[70] M. Fisher, M. Singh, D. Spears, and M. Wooldridge. Guest Editorial: Logic-Based Agent Verification. *Journal of Applied Logic*, 5(2):193–195, 2007.

[71] M. Fisher and M. Wooldridge. On the Formal Specification and Verification of Multi-Agent Systems. *International Journal of Cooperative Information Systems*, 6(1):37–65, Jan. 1997.

[72] S. Franklin and A. Graesser. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In *Intelligent Agents III (Proc. 3rd International Workshop on Agent Theories, Architectures, and Languages)*, volume 1193 of *LNCS*, pages 21–35. Springer, 1996.

[73] M. P. Georgeff and A. L. Lansky. Reactive Reasoning and Planning. In *Proc. 6th National Conference on Artificial Intelligence (AAAI)*, pages 677–682, 1987.

[74] L. Giordano and A. Martelli. Verifying Agents' Conformance with Multiparty Protocols. In *Proc. 9th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 5405 of *LNCS*, pages 17–36. Springer, 2008.

[75] L. Giordano, A. Martelli, and C. Schwind. Specifying and Verifying Interaction Protocols in a Temporal Action Logic. *Journal of Applied Logic*, 5(2):214–234, 2007.

[76] F. Giunchiglia and L. Serafini. Multilanguage Hierarchical Logics (or: how we can do without modal logics). *Artificial Intelligence*, 65:29–70, 1994.

[77] T. F. Gordon. Constructing Legal Arguments with Rules in the Legal Knowledge Interchange Format (LKIF). In *Computable Models of the Law, Languages, Dialogues, Games, Ontologies*, volume 4884 of *LNCS*, pages 162–184. Springer, 2008.

[78] P. Gouveia and J. Ramos. Multi-Agent Systems Specification and Certification: A Situation and State Calculus Approach. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):301–338, 2004.

[79] H. Hansson and B. Jonsson. A Logic for Reasoning about Time and Reliability. *Formal Aspects of Computing*, 6:102–111, 1994.

[80] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.

[81] J. Harland and M. Winikoff. Agents via Mixed-Mode Computation in Linear Logic. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):167–196, 2004.

[82] M. He and N. R. Jennings. Southampton TAC: An Adaptive Autonomous Trading Agent. *ACM Transactions on Internet Technology*, 3(3):218–235, 2003.

[83] K. V. Hindriks, F. S. de Boer, W. van der Hoek, and J.-J. Meyer. Agent Programming with Declarative Goals. In *Intelligent Agents VII (Proc. 6th Workshop on Agent Theories, Architectures, and Languages — ATAL)*, volume 1986 of *LNAI*, pages 228–243. Springer, 2001.

[84] J. Huang, R. L. Z. Gutierrez, B. M. García, and M. N. Huhns. Reconciling Agent Ontologies for Web Service Applications. In *Proc. 3rd German Conference on Multiagent System Technologies (MATES)*, volume 3550 of *LNCS*, pages 106–117. Springer, 2005.

[85] U. Hustadt, C. Dixon, R. A. Schmidt, M. Fisher, J.-J. Meyer, and W. van der Hoek. Verification Within the KARO Agent Theory. In *Agent Technology from a Formal Perspective*, NASA Monographs in Systems and Software Engineering. Springer, 2006.

[86] K. Inoue and K. Iwanuma. Speculative Computation Through Consequence-Finding in Multi-Agent Environments. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):255–291, 2004.

[87] W. Jamroga and J. Dix. Model Checking Abilities of Agents: A Closer Look. *Theory of Computing Systems*, 42(3):366–410, 2008.

[88] N. R. Jennings and M. Wooldridge. Applications of Agent Technology. In *Agent Technology: Foundations, Applications, and Markets*. Springer, 1998.

[89] S.-S. T. Q. Jongmans, K. V. Hindriks, and M. B. van Riemsdijk. Model Checking Agent Programs by Using the Program Interpreter. In *Proc. 11th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 6245 of *LNCS*, pages 219–237. Springer, 2010.

[90] A. C. Kakas, P. Mancarella, F. Sadri, K. Stathis, and F. Toni. Computational Logic Foundations of KGP Agents. *Journal of Artificial Intelligence Research*, 33:285–348, 2008.

[91] K. Konolige. A First-Order Formalization of Knowledge and Action for a Multi-Agent Planning System. In J. E. Hayes, D. Michie, and Y. Pao, editors, *Machine Intelligence 10*, pages 41–72. Ellis Horwood, 1982.

[92] R. A. Kowalski and F. Sadri. From Logic Programming Towards Multi-Agent Systems. *Annals of Mathematics and Artificial Intelligence*, 25(3-4):391–419, 1999.

[93] S. Kraus. Beliefs, Time and Incomplete Information in Multiple Encounter Negotiations Among Autonomous Agents. *Annals of Mathematics and Artificial Intelligence*, 20(1-4):111–159, 1997.

[94] A. Kurucz. Combining Modal Logics. In Blackburn et al. [16], pages 869–924.

[95] F. Laroussinie, N. Markey, and G. Oreiby. On the Expressiveness and Complexity of ATL. *Logical Methods in Computer Science*, 4(2), 2008.

[96] V. Lesser, C. L. Ortiz, and M. Tambe, editors. *Distributed Sensor Networks: A Multiagent Perspective*. Kluwer Academic Publishers, 2003.

[97] H. Levesque, R. Reiter, Y. Lespérance, F. Lin, and R. Scherl. GOLOG: A Logic Programming Language for Dynamic Domains. *Journal of Logic Programming*, 31(1-3):59–83, 1997.

[98] A. Lomuscio, H. Qu, and F. Raimondi. MCMAS: A Model Checker for the Verification of Multi-Agent Systems. In *Proc. 21st International Conference on Computer Aided Verification (CAV)*, volume 5643 of *LNCS*, pages 682–688. Springer, 2009.

[99] MCMAS: a Model Checker for Multi-Agents Systems. `http://www-lai.doc.ic.ac.uk/mcmas`.

[100] R. C. Moore. *Logic and Representation (CSLI Lecture Notes Number 39)*. Center for the Study of Logic and Information (CSLI), Stanford University, USA, 1995. (Distributed by Chicago University Press).

[101] S. A. Moore. KQML & FLBC: Contrasting Agent Communication Languages. In *HICSS*, 1999.

[102] C. Muldoon, G. M. P. O'Hare, R. W. Collier, and M. J. O'Grady. Agent Factory Micro Edition: A Framework for Ambient Applications. In *Proc. 6th International Conference on Computational Science (ICCS); Part III*, volume 3993 of *LNCS*, pages 727–734. Springer, 2006.

[103] N. Muscettola, P. P. Nayak, B. Pell, and B. Williams. Remote Agent: To Boldly Go Where No AI System Has Gone Before. *Artificial Intelligence*, 103(1-2):5–48, 1998.

[104] M. Nguyen-Duc, Z. Guessoum, O. Marin, J.-F. Perrot, J.-P. Briot, and V. Duong. Towards a Reliable Air Traffic Control. In *Proc. 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), Industry and Applications Track*, pages 101–104, 2008.

[105] G. Nitschke. Cooperating Air Traffic Control Agents. *Applied Artificial Intelligence*, 15(2):209–235, 2001.

[106] J. Odelstad and M. Boman. Algebras for Agent Norm-Regulation. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):141–166, 2004.

[107] J. A. Padget, A. Artikis, W. W. Vasconcelos, K. Stathis, V. T. da Silva, E. T. Matson, and A. Polleres, editors. *Proc. 5th International Workshop on Coordination, Organizations, Institutions and Norms in Agent Systems (COIN)*, volume 6069 of *LNCS*. Springer, 2010.

[108] A. Pnueli. The Temporal Logic of Programs. In *Proc, 18th Symposium on the Foundations of Computer Science (FOCS)*, pages 46–57. IEEE Computer Society Press, 1977.

[109] A. Pokahr, L. Braubach, and W. Lamersdorf. Jadex: A BDI Reasoning Engine. In Bordini et al. [18], chapter 6, pages 149–174.

[110] I. Rahwan, P. Pasquier, L. Sonenberg, and F. Dignum. A Formal Analysis of Interest-Based Negotiation. *Annals of Mathematics and Artificial Intelligence*, 55(3-4):253–276, 2009.

[111] F. Raimondi and A. Lomuscio. Automatic Verification of Multi-agent Systems by Model Checking via Ordered Binary Decision Diagrams. *Journal of Applied Logic*, 5(2):235–251, 2007.

[112] A. Rao. AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language. In *Agents Breaking Away: Proc. 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW)*, volume 1038 of *LNCS*, pages 42–55. Springer, 1996.

[113] A. S. Rao. Decision Procedures for Propositional Linear-Time Belief-Desire-Intention Logics. *Journal of Logic and Computation*, 8(3):293–342, 1998.

[114] A. S. Rao and M. Georgeff. BDI Agents: from Theory to Practice. In *Proc. 1st International Conference on Multi-Agent Systems (ICMAS)*, pages 312–319, 1995.

[115] A. S. Rao and M. P. Georgeff. Modeling Agents within a BDI-Architecture. In *Proc. 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 473–484. Morgan Kaufmann, 1991.

[116] A. S. Rao and M. P. Georgeff. An Abstract Architecture for Rational Agents. In *Proc. International Conference on Knowledge Representation and Reasoning (KR&R)*, pages 439–449, 1992.

[117] A. S. Rao and M. P. Georgeff. A Model-Theoretic Approach to the Verification of Situated Reasoning Systems. In *Proc. 13th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 318–324, 1993.

[118] A. Rogers, D. D. Corkill, and N. R. Jennings. Agent Technologies for Sensor Networks. *IEEE Intelligent Systems*, 24(2):13–17, 2009.

[119] S. Sardiña, G. D. Giacomo, Y. Lespérance, and H. J. Levesque. On the Semantics of Deliberation in Indigolog - from Theory to Implementation. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):259–299, 2004.

[120] A. Sayda and J. H. Taylor. Toward A Practical Multi-agent System for Integrated Control and Asset Management of Petroleum Production Facilities. In *Proc. IEEE International Symposium on Intelligent Control*, 2007.

[121] K. Schild. On the Relationship Between BDI Logics and Standard Logics of Concurrency. In *Intelligent Agents V (Proc. 5th International Workshop on Agent Theories, Architectures, and Languages — ATAL)*, volume 1555 of *LNAI*, pages 47–61. Springer, 1999.

[122] R. A. Schmidt and D. Tishkovsky. Multi-Agent Dynamic Logics with Informational Test. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):5–36, 2004.

[123] O. Schulte and J. P. Delgrande. Representing von Neumann-Morgenstern Games in the Situation Calculus. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):73–101, 2004.

[124] J. R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[125] A. A. Sherstov and P. Stone. Three Automated Stock-Trading Agents: A Comparative Study. In *Proc. 6th Workshop on Agent-Mediated Electronic Commerce (AMEC)*, volume 3435 of *LNCS*, pages 173–187. Springer, 2005.

[126] J. S. Sichman, J. A. Padget, S. Ossowski, and P. Noriega, editors. *Proc. 3rd International Workshop on Coordination, Organizations, Institutions and Norms in Agent Systems (COIN)*, volume 4870 of *LNCS*. Springer, 2008.

[127] M. P. Singh. A Semantics for Speech Acts. *Annals of Mathematics and Artificial Intelligence*, 8(1-2):47–71, 1993.

[128] V. S. Subrahmanian, P. Bonatti, J. Dix, T. Eiter, S. Kraus, F. Ozcan, and R. Ross. *Heterogeneous Agent Systems: Theory and Implementation*. MIT Press, 2000.

[129] W. van der Hoek, A. Lomuscio, and M. Wooldridge. On the Complexity of Practical ATL Model Checking. In *Proc. 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 201–208. ACM Press, 2006.

[130] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library Series*. Springer, 2007.

[131] B. van Linder, W. van der Hoek, and J. J. C. Meyer. Formalising Motivational Attitudes of Agents. In *Intelligent Agents II (Proc. 2nd Workshop on Agent Theories, Architectures, and Languages — ATAL)*, volume 1037 of *LNCS*, pages 17–32. Springer, 1996.

[132] D. Van Nieuwenborgh, M. De Vos, S. Heymans, and D. Vermeir. Hierarchical Decision Making in Multi-agent Systems Using Answer Set Programming. In *Proc. 7th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA)*, volume 4371 of *LNCS*, pages 20–40. Springer, 2007.

[133] W. W. Vasconcelos, D. S. Robertson, C. Sierra, M. Esteva, J. Sabater, and M. Wooldridge. Rapid Prototyping of Large Multi-Agent Systems Through Logic Programming. *Annals of Mathematics and Artificial Intelligence*, 41(2-4):135–169, 2004.

[134] R. Vieira, Á. F. Moreira, M. Wooldridge, and R. H. Bordini. On the Formal Semantics of Speech-Act Based Communication in an Agent-Oriented Programming Language. *Journal of Artificial Intelligence Research*, 29:221–267, 2007.

[135] W. Visser, K. Havelund, G. Brat, and S. Park. Model Checking Programs. In *Proc. 15th IEEE International Conference on Automated Software Engineering (ASE)*, pages 3–12. IEEE Computer Society Press, 2000.

[136] M. Winikoff. JACK^{TM} intelligent agents: An industrial strength platform. In Bordini et al. [18], chapter 7, pages 175–193.

[137] M. Wooldridge. *Reasoning about Rational Agents*. MIT Press, 2000.

[138] M. Wooldridge, M. Fisher, M.-P. Huget, and S. Parsons. Model Checking for Multiagent Systems: The MABLE Language and its Applications. *International Journal of Artificial Intelligence Tools*, 15(2):195–225, 2006.

[139] M. Wooldridge and N. R. Jennings. Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.

[140] M. Wooldridge and A. Rao, editors. *Foundations of Rational Agency*. Applied Logic Series. Kluwer Academic Publishers, 1999.

[141] L. Xiao, P. H. Lewis, and S. Dasmahapatra. Secure Interaction Models for the HealthAgents System. In *Proc. 27th International Conference on Computer Safety, Reliability, and Security (SAFECOMP)*, volume 5219 of *LNCS*, pages 167–180. Springer, 2008.

[142] P. Yolum and M. P. Singh. Reasoning about Commitments in the Event Calculus: An Approach for Specifying and Executing Protocols. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):227–253, 2004.